

Claims

WHAT IS CLAIMED IS:

- 1 1. A computing system having a mass storage device and a system timer for
2 obtaining benchmark timing for a portion of an application program execution, the computing
3 system comprising:
4 a mass storage system;
5 an init module for determining if the timestamp data is to be collected during the
6 operation of the application program;
7 a performance marker module for obtaining and storing the timestamp data for later
8 retrieval;
9 an uninit module for formatting and storing the obtained timestamp data into a data file
10 within the mass storage device that permits retrieval after the termination of the application
11 program; and
12 a performance benchmark data post processing module for determining the benchmark
13 timing from two or more timestamp data entries;
14 wherein
15 the init module is executed before any timestamp data is collected;
16 the performance marker module is executed each time benchmark timestamp data
17 and overhead timestamp data is to be collected;
18 the uninit module is executed after all timestamp data desired has been collected;
19 and
20 the performance benchmark data post processing module determines the
21 benchmark timing from timestamp entries stored within the data file.

1 2. The computing system according to claim 1, wherein the init module determines if
2 timestamp data is to be collected.

1 3. The computing system according to claim 2, wherein init module makes the
2 determination that timestamp data is to be collected by checking for the existence of an
3 identification key within a system registry;
4 the identification key uniquely identifying the processing modules to be used to collect,
5 format, and store the run-time internal state data to be collected.

1 4. The computing system according to claim 3, wherein the timestamp data
2 comprises a timer count value obtained from the system timer.

1 5. The computing system according to claim 2, wherein the performance marker
2 module collects timestamp data only if the init module has determined that the timestamp data is
3 to be collected.

1 6. The computing system according to claim 5, wherein the performance marker
2 module generates a benchmark data record containing a benchmark timestamp data value each
3 time the performance marker module is executed.

1 7. The computing system according to claim 6, wherein the benchmark data record
2 further containing an overhead timestamp data value each time the performance marker module is
3 executed.

1 8. The computing system according to claim 7, wherein the performance marker
2 module stores the benchmark data records within a data memory block within the processing
3 modules identified by an identification key within a system registry.

1 9. The computing system according to claim 8, wherein a unit module retrieves
2 the benchmark data records from the data memory block for transfer to the data file on the mass
3 storage device.

1 10. The computing system according to claim 9, wherein the performance benchmark
2 data post processing module determines the benchmark timing from difference between two
3 benchmark timestamp data entries stored within the data file.

1 11. The computing system according to claim 10, wherein the performance
2 benchmark data post processing module determines the benchmark timing by subtracting an
3 estimate for the total overhead processing from the difference between two benchmark timestamp
4 data entries stored within the data file.

1 12. The computing system according to claim 11, wherein the estimate for the total
2 overhead processing is determined by totaling the difference between the overhead timestamp
3 value and the benchmark timestamp value for all code markers between the two benchmark
4 timestamp entries used to determine the benchmark timing.

1 13. A method for obtaining benchmark timing for a portion of an application program
2 execution, the method comprising:

3 inserting one or more code markers into the application program at locations within the
4 application program corresponding to the point at which benchmark timing data is desired;

5 determining if benchmark timing data is to be collected at each code marker by checking
6 for the existence of processing modules identified by an identification key within a system
7 registry;

8 if benchmark timing data is to be collected at each code marker:

9 generating benchmark data record containing the collected benchmark timing
10 data each time the code markers are reached;
11 storing the benchmark data records within a data memory block within the
12 processing modules identified by the identification key within the system registry;
13 retrieving the benchmark data records from the data memory block for transfer to
14 a mass storage device once all of the run-time internal state data has been collected; and
15 processing the benchmark data records stored within the mass storage device to
16 determine the benchmark timing defined between two benchmark data records.

1 14. The method according to claim 13, wherein the benchmark timing from difference
2 between two benchmark timestamp data entries stored within the data file.

1 15. The method according to claim 14, wherein the benchmark timing is determined
2 by subtracting an estimate for the total overhead processing from the difference between two
3 benchmark timestamp data entries stored within the data file.

1 16. The method according to claim 15, wherein the estimate for the total overhead
2 processing is determined by totaling the difference between an overhead timestamp value and a
3 benchmark timestamp value for all code markers between the two benchmark timestamp entries
4 used to determine the benchmark timing.

1 17. The method according to claim 16, wherein
2 the benchmark timestamp value is obtained from a system timer immediately after a code
3 marker is reached;
4 the overhead timestamp value is obtained from the system timer immediately before
5 processing returns to the application program from performance marker processing.

1 18. A computer data product readable by a computing system and encoding a
2 computer program of instructions for executing a computer process for obtaining run-time
3 internal state data within an application program, said computer process comprising the steps of:
4 inserting one or more code markers into the application program at locations within the
5 application program corresponding to the point at which benchmark timing data is desired;
6 Determining if benchmark timing data is to be collected at each code marker by checking
7 for the existence of processing modules identified by an identification key within a system
8 registry;
9 if benchmark timing data is to be collected at each code marker:
10 generating a benchmark data record containing the collected benchmark timing
11 data each time the code markers are reached;
12 storing the benchmark data records within a data memory block within the
13 processing modules identified by the identification key within the system registry;
14 retrieving the benchmark data records from the data memory block for transfer to
15 a mass storage device once all of the run-time internal state data has been collected; and
16 processing the benchmark data records stored within the mass storage device to
17 determine the benchmark timing defined between two benchmark data records.

1 19. The computer data product according to claim 18, wherein the determining step
2 makes the determination that benchmark timing data is to be collected by checking for the
3 existence of an identification key within a system registry;
4 the identification key uniquely identifies the processing modules to be used to collect,
5 format, and store the run-time internal state data to be collected.

1 20. The computer data product according to claim 19, wherein the determining step
2 further makes the determination that benchmark timing data is to be collected by checking for the
3 existence of processing modules identified by the identification key within the system registry.

1 21. The computer data product according to claim 19, wherein the data memory block
2 is within processing modules identified by the identification key within the system registry.

1 22. The computer data product according to claim 21, wherein the benchmark timing
2 from difference between two benchmark timestamp data entries stored within the data file.

1 23. The computer data product according to claim 22, wherein the benchmark timing
2 is determined by subtracting an estimate for the total overhead processing from the difference
3 between two benchmark timestamp data entries stored within the data file.

1 24. The computer data product according to claim 23, wherein the estimate for the
2 total overhead processing is determined by totaling the difference between an overhead
3 timestamp value and a benchmark timestamp value for all code markers between the two
4 benchmark timestamp entries used to determine the benchmark timing.

1 25. The computer data product according to claim 24, wherein
2 the benchmark timestamp value is obtained from a system timer immediately after a code
3 marker is reached; and

4 the overhead timestamp value is obtained from the system timer immediately before
5 processing returns to the application program from performance marker processing.

1 26. The computer data product according to claim 19, wherein the computer data
2 product comprises a computer readable storage medium readable by a computer upon which
3 encoded instructions used to implement the computer process are stored.

27. The computer data product according to claim 19, wherein the computer data product comprises a propagated signal on a carrier detectable by a computing system and encoding a computer program of instructions for executing the computer process.

THE